# Conteúdo

Para que precisamos de robôs? p. 142

Robôs da fischertechnik p. 144
- Atuadores p. 144
- Sensores p. 144
- Interface ROBO p. 145
- Software ROBO Pro p. 145
- Alimentação de corrente p. 145
- Procedimento em experiências p. 146

Primeiros passos p. 146

O primeiro robô simples p. 148

Robôs rolantes inteligentes p. 150
- Modelo básico p. 150
- O robô que segue um foco de luz p. 152
- O robô que segue uma trilha p. 154
- Robô com detecção de obstáculos p. 155
- Robô que segue um foco de luz com detecção de obstáculos p. 158
- Robô com detecção de bordas p. 160

O robô andante p. 163

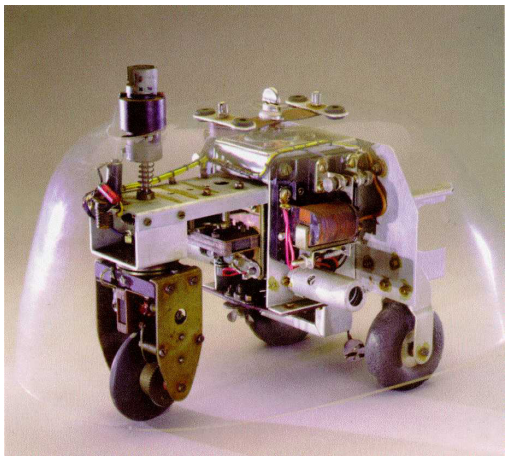
Possibilidades de expansão p. 165
- Emissor manual de infravermelhos p. 165
- ROBO RF Data Link p. 165
- ROBO I/O-Extension p. 166

Busca de falhas p. 167

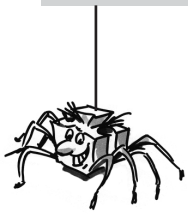
# Para que precisamos de robôs?

■ O termo Robô foi utilizado pela primeira vez em 1923 no romance "Golem" de Carel Capek. Esta figura criada no meio artístico devia, com as suas características, substituir o trabalho humano.
Nos anos 30 e 40 do século XX, a partir do robô surgiu uma arte robótica. Diversas tentativas de lhe fornecer características humanas, por exemplo uma cabeça com olhos formados por lâmpadas a piscar, nos provocam ainda atualmente um sorriso cansado. Mobilidade ou até mesmo inteligência ainda se nota muito pouco nestas máquinas. Como o princípio do controle tem grande influência na técnica robótica, a chegada de conexões eletrónicas tornaram a construção de robôs mais realista. A questão da "inteligência" de robôs é ainda hoje objeto de pesquisa e investigação de muitas firmas, institutos e universidades.

■ As primeiras aproximações ao problema foram esperadas da denominada cibernética. A designação "cibernética" deriva da palavra grega Kybernetes. Kybernetes era o navegador de barcos a remos gregos. Ele tinha de determinar o local do barco e alcançar a rota necessária até ao destino.
Com isso fica claro que a cibernética deveria tornar o robô "inteligente". Como se pode representar tal comportamento inteligente?
Nós queremos tentar esclarecer este fato com a ajuda de uma experiência de pensamento. Todos já observamos o comportamento de uma mariposa no círculo de luz de uma lâmpada. A mariposa detecta a fonte de luz, voa para lá, e a seguir se afasta pouco antes de bater na lâmpada. Através deste comportamento se torna evidente que a mariposa detecta a fonte de luz, pesquisa um caminho até ela e, depois, tem de voar para lá. Estas capacidades se baseiam em modelos de comportamento instintivos, inteligentes do inseto.
Tentamos transferir agora estas capacidades para um sistema técnico. Temos de reconhecer a fonte de luz (sensores ópticos), executar um movimento (comandar motores) e temos de estabelecer uma relação conveniente entre reconhecimento e movimento (o programa).



■ A experiência de pensamento descrita anteriormente foi realizada pelo inglês Walter Grey nos anos 50.
Com a ajuda de sensores simples, motores e conexões eletrônicas foram conseguidos diferentes animais "cibernéticos" que adotaram comportamentos muito específicos, como por exemplo o de uma mariposa. A figura mostra uma réplica da tartaruga "cibernética" que está exposta no Museu Smithsonian em Washington

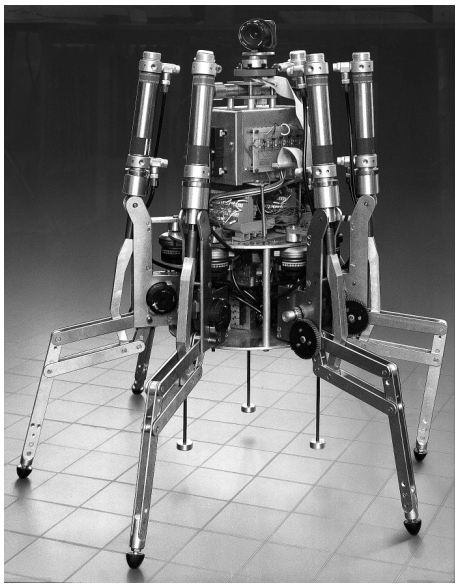


■ Mas para que precisamos de robôs móveis? Uma vez tentamos aplicar o comportamento da nossa "mariposa de pensamento" a aparelhos técnicos. Um exemplo simples é a busca de luz. Modificamos a fonte de luz, colocando uma tira clara, a linha guia, no chão e os sensores deixam de estar orientados para a frente e sim para baixo. Com a ajuda das linhas guia desse tipo, um robô móvel pode se orientar, por exemplo, num hall de armazém. Informações adicionais, por exemplo na forma de código de barras em determinados locais, levam o robô nestas posições a outras ações, como por exemplo a recolha ou depósito de uma palete.

Tais sistemas robóticos já existem na realidade. Em grandes hospitais existem longas vias de transporte para materiais de consumo, como por exemplo roupas de cama. O transporte destes materiais pelo pessoal da manutenção é dispendioso e está ligado, em parte, a trabalho físico intenso. Além disso, tais atividades se limitam ao tempo existente para a manutenção dos pacientes.

■ Há alguns anos, os cientistas se ocuparam com uma outra forma de movimento muito generalizada na natureza, o andar e o correr. Foram desenvolvidos robôs que conseguiam se movimentar com pernas. Um exemplo de um robô andante de seis pernas é o robô andante eletropneumático "Achille" desenvolvido na academia real militar em Bruxelas. Equipado com uma câmara no topo e nas seis pernas, este robô deve reagir mecanicamente a obstáculos elevados ou profundos (objetos ou buracos).

Tais máquinas andantes podem ser aplicadas onde os veículos de rodas e de lagarta não têm qualquer chance, como por exemplo em terrenos exteriores irregulares ou flexíveis, no caso ultrapassar obstáculos, subir escadas, passar valas ou na aplicação em pontos perigosos e de difícil acesso em centrais nucleares, galerias de minas ou em ações de salvamento.



Reconhecemos também que o robô móvel pode ocupar um lugar importante na sociedade moderna.

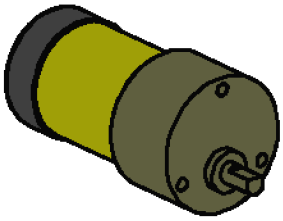
# Robôs da fischertechnik

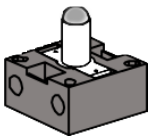
■ Como podemos construir robôs a partir dos nossos módulos da fischertechnik? Para um robô precisamos, além dos sensores (por exemplo botões) e atuadores (por exemplo motores), de muitas peças para construir um modelo. O módulo da fischertechnik ROBO Mobile Set é a base ideal. Os seguintes sensores e atuadores estão incluídos neste módulo:

## Atuadores

**Motor power:**

Dois destes motores em série de corrente contínua (9VDC/2,4W) com caixa de velocidades incorporada e uma redução de 50:1 acionam os modelos de robôs móveis (i.e. o motor roda 50 vezes à volta do veio, que se salienta do motor, no mesmo tempo apenas uma vez).





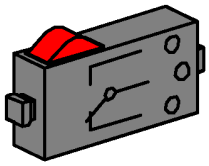
**Lâmpada de formato lenticular:**

Para a edição de sinais de luz mais simples é apropriada a lâmpada incandescente (9VDC/150mA). Na ampola de vidro da lâmpada está integrada uma lente que enfaixa a luz que sai. Se o feixe de luz for direcionado para um sensor de luminosidade (fototransístor, ver em baixo) pode se montar uma barreira fotoeléctrica que distingue a luminosidade da escuridão. A lâmpada pode também ser utilizada para indicar determinados objetos ou como lâmpada a piscar para emissão de mensagens de aviso. No módulo é utilizada a lâmpada juntamente com 2 fototransístores como sensor especial para reconhecimento da linha.

## Sensores

■ No caso de sensores se distingue entre sensores digitais e analógicos.

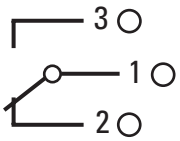
O **botão** é um exemplo de um sensor digital. Dimensões digitais podem aceitar 2 estados diferentes. Estes estados são designados por 0 ou 1. Para o botão, o "0" significa que não passa qualquer corrente entre as ligações, "1" significa que há corrente.

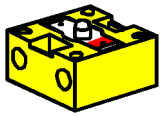


O botão da fischertechnik está concebido como seletor. Por isso, possui 3 ligações. Ao premir o botão vermelho é acionado mecanicamente um interruptor que liga as conexões 1 e 3 entre si. Simultaneamente, o contato é interrompido entre as ligações 1 e 2 que estavam ligadas no estado de repouso. Desta forma, podem ser consultadas ambas as posições iniciais possíveis:

No estado de repouso fechado (ligações 1 e 2 ocupadas)

No estado de repouso aberto (ligações 1 e 3 ocupadas)



O **fototransístor** pode ser utilizado como sensor digital assim como sensor analógico. No primeiro caso, ele serve para a detecção da passagem nítida claro-escuro, p. ex. de uma linha selecionada. As quantidades de luz se podem distinguir pela sua intensidade, nesse caso o fototransístor funciona como sensor analógico. Os valores analógicos podem ser alterados como quiser entre os seus valores limite. Para que estas grandezas possam ser processadas por computador, têm de ser convertidas em valores numéricos correspondentes.

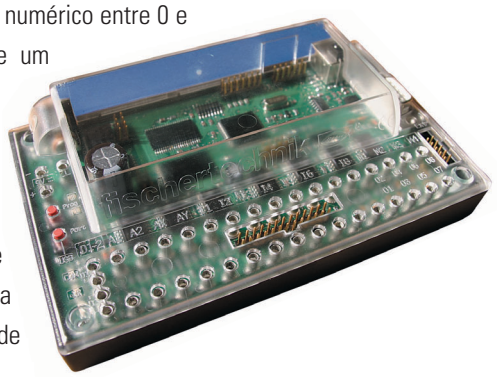
No caso do fototransístor, se trata normalmente de um chamado dispositivo semicondutor, cujas propriedades elétricas estão dependentes da luminância. Todos conhecem as células solares com as

quais se pode produzir corrente a partir da luz solar. Podemos entender o fototransístor como uma combinação de mini células solares e transístor. Os impulsos de luz que incidem no fototransístor (fotões) criam uma corrente muito pequena que é fortalecida pelo transístor.

**Nota:**
Na ligação do fototransístor prestar atenção à polaridade correta: Marcação vermelha = positivo.
Tensão permitida: 30V máx.

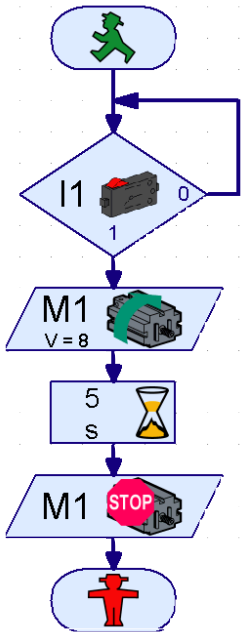
## ROBO Interface

■ No ROBO Interface podemos ligar e avaliar diferentes sensores e atuadores. Além das 8 entradas digitais, o ROBO Interface dispõe de outras entradas analógicas. Assim, por exemplo, um valor de resistência aplicado nas entradas AX e AY entre 0 e 5,5 kΩ é convertido num valor numérico entre 0 e 1024. Os valores de medição de um sensor de luminosidade, por exemplo de um fototransístor, são recolhidos e estão disponíveis para um outro processamento. Nas entradas analógicas A1 e A2 podem ser medidas tensões entre 0 e 10VDC.



A função mais importante da interface consiste na combinação lógica de grandezas de entrada. Para isso, a interface necessita de um programa. O programa decide que tipo de dados de entrada são considerados os sinais do sensor, os dados de saída adequados, sinais de controle do motor etc.. Com o ROBO Interface dispomos de suficiente potência de processamento de dados para projetar programas exigentes.

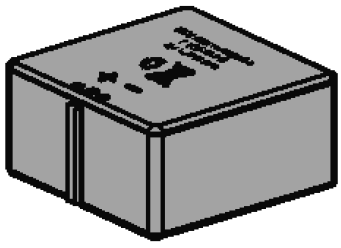
## Software ROBO Pro




■ Com isso é efetivamente possível conseguir criar os programas necessários para a interface, pois existe uma superfície gráfica de programação. Por trás do termo "superfície gráfica de programação" se esconde um software que nos possibilita criar os nossos programas de uma forma muito confortável. Isto acontece com a ajuda de símbolos gráficos. O computador do ROBO Interface pode executar apenas comandos a partir do seu conjunto de comandos conhecido. Essencialmente, são estruturas de controle simples, cujas aplicações são extraordinariamente difíceis para os principiantes. O software do PC do ROBO Pro coloca, por isso, à disposição elementos gráficos que são traduzidos, posteriormente num idioma executável para a interface.

## Alimentação de corrente

■ A única coisa de que ainda necessitamos para o módulo ROBO Mobile Set, é o Accu Set art. n.º 34969. Contém o conjunto de baterias como alimentação de corrente móvel para os nossos modelos de robôs e um carregador especial para o conjunto de baterias.



O melhor será carregar a bateria logo com o carregador, para que esteja carregada quando desejarmos começar as experiências.

## Procedimento em experiências

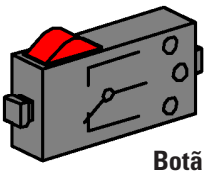
■ Na nossa entrada no fascinante mundo dos robôs móveis iremos avançar passo a passo. Começamos com uma montagem de teste simples para a verificação das funções básicas da interface e sensor. Em seguida, montamos modelos simples cujas tarefas estabelecidas estão atribuídas e procuramos sempre sistemas mais complexos.
Quando a criação de programas próprios se torna demasiado complexa e demora demasiado tempo, então se pode carregar os programas de exemplo fornecidos na interface e acionar com eles o robô.
Para não desesperar quando ocorrerem erros, existe no final um capítulo sobre busca de falhas.

Um ponto muito importante é o cuidado na montagem e na colocação em funcionamento dos nossos robôs. Na ligação dos componentes elétricos seguimos escrupulosamente as instruções e verificamos duas ou três vezes se tudo estava em condições. No caso de construções mecânicas, assim como no caso de criações próprias, prestamos muita atenção à mobilidade e à folga reduzida nas caixas de velocidades e fixações. A nossa criatividade está reservada para escrever programas próprios e definir um novo "comportamento". Só está limitada à quantidade de memória e capacidade de processamento de dados do hardware. Os seguintes exemplos fornecem alguns estímulos.

## Primeiros passos

■ Após as reflexões teóricas pretendemos agora começar a efetuar algumas experiências. Certamente que vai querer experimentar imediatamente, talvez até com um robô andante grande. Isso é possível e, prestando uma cuidadosa atenção às instruções de montagem, a montagem do modelo se consegue logo na primeira tentativa.
Mas o que fazer quando não funciona? Nestes casos, tem de se tentar encontrar sistematicamente causas para os erros. Antes de nos ocuparmos disso, verificamos a combinação do computador e interface.

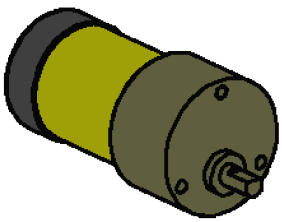
No manual de instruções do ROBO Pro é descrito no capítulo 1 e 2, como o software de controle é instalado no PC e como a interface é conectada. Com a ajuda do teste de interface testamos os diferentes sensores e atuadores.



Botã

**Botã**

Podemos agora, por exemplo, ligar um botão na entrada digital I1 e observar a alteração do estado da entrada ao acionar o botão.



Motor power

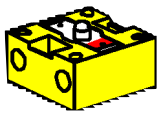
**Motor power**

Verificamos as saídas ao mesmo tempo que ligamos um motor com uma saída de motor, por exemplo M1. Com o botão esquerdo do mouse, podemos colocar o motor em rotação e alterar a velocidade com a válvula corrediça.

**Fototransístor**

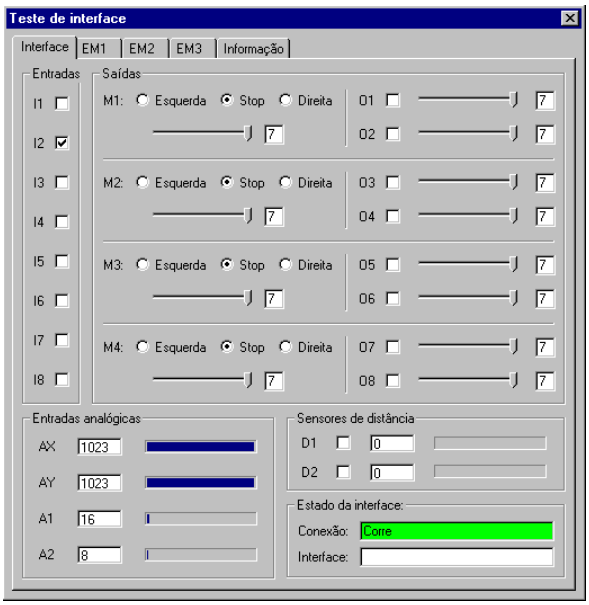
Se pretendermos testar também a entrada analógica AX, podemos utilizar um fototransístor como sensor analógico.

Enquanto que no caso do motor ou do botão a polaridade não é relevante (o motor funciona na pior das hipóteses em rotação contrária), a ligação correta do fototransístor é extremamente importante para o funcionamento correto.
Ligamos o contato do transístor com a marcação vermelha com um conector adaptador, o outro contato com um conector verde. O segundo conector verde encaixa no plugue da entrada AX que se encontra perto da margem da interface, o segundo conector vermelho se adapta na plugue de AX que se encontra bem no interior. (atenção: Se se fechar o fototransístor na entrada digital I1-I8, o conector vermelho encaixa na plugue que se encontra perto da margem da caixa).

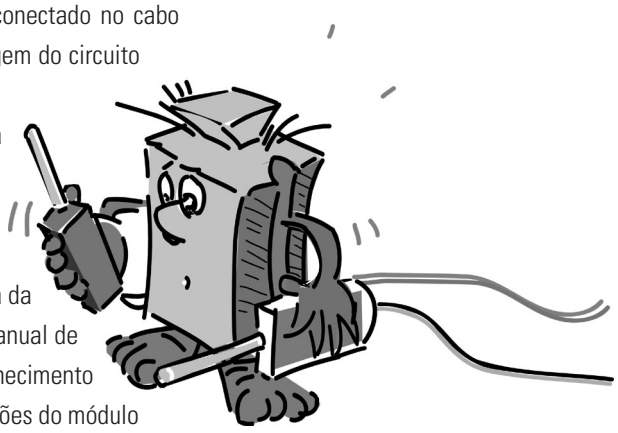


**Fototransistor**

Agora podemos variar a intensidade da luminosidade do transístor com a ajuda de uma lanterna de bolso e alterar a amplitude da viga azul de AX. Se o indicador não se mover em caso de uma amplitude máxima, volte a observar as ligações do fototransístor. Se, pelo contrário, o indicador estiver em zero com a lâmpada de bolso desligada, então pode ser que a iluminação na sala, a luminosidade do ambiente, seja demasiado elevada. A amplitude se altera quando destapamos o fototransístor.



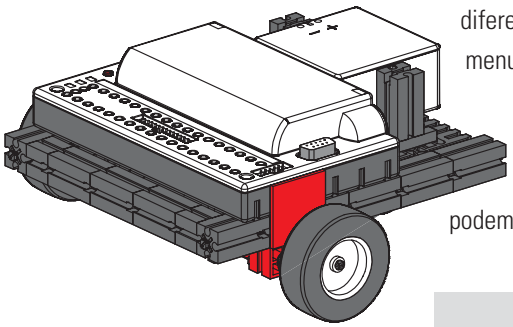

Voltando novamente ao assunto da distribuição de cores dos conectores: Prestamos sempre atenção para que, na montagem, um conector vermelho seja conectado no cabo vermelho e para que um conector verde seja conectado no cabo verde. Se a montagem do circuito depender da polaridade correta, colocamos sempre um cabo vermelho para o pólo positivo e um verde para o negativo. Pode parecer um pouco meticuloso, mas para uma busca de falhas sistemática, uma atribuição clara de cores torna tudo consideravelmente mais fácil.



Com um programa simples pretendemos concluir os nossos primeiros passos na área da robótica. O programa "controle do portão da garagem" explicado no capítulo 3 do manual de instruções ROBO Pro não tem a ver com robôs móveis, sendo mais adequado ao conhecimento do software ROBO Pro. Para poder executar o programa basta ligar o motor e três botões do módulo ROBO Mobile Set à interface. Tudo o resto é descrito pormenorizadamente no manual de instruções do software.

# O primeiro robô simples

■ Após o teste de interface e o controle do portão da garagem pretendemos, por fim, colocar o primeiro robô em funcionamento. Montamos o modelo "robô simples" com ambos os motores de acionamento de acordo com as instruções de montagem. Isso é relativamente fácil e rápido, porque este modelo contém apenas o necessário que um robô precisa para andar. Ligamos os motores nas saídas M1 e M2.



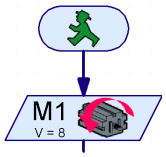
Nós acedemos ao ROBO Pro e colocamos um novo programa (ARQUIVO – NOVO). ROBO Pro oferece diferentes níveis de dificuldade em que podemos trabalhar. No ROBO Pro pode configurar o item do menu LEVEL. Para já, basta-nos o Level 1.

Surge uma folha de trabalho vazia e na margem esquerda da tela a janela elementar, na qual podemos selecionar diferentes elementos de programa com o botão esquerdo do mouse e podemos colocar na superfície de trabalho. Com o botão direito do mouse alteramos as propriedades.

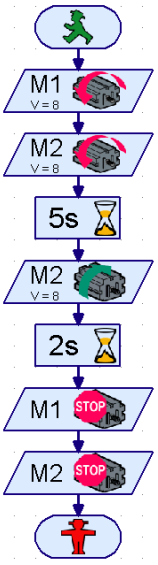
**Tarefa 1 (Level 1):**
**O nosso "robô simples" deve andar em frente por 5 segundos, a seguir em circuito por 2 segundos e, depois, permanecer parado.**

**Dicas:**

Programamos o primeiro robô passo a passo em conjunto:




- Começamos com o semáforo pedestre em movimento. Simboliza o início do programa.
- Em primeiro lugar, buscamos o símbolo do motor da janela do elemento e colocámo-lo por baixo do elemento inicial de forma que a linha de conexão é indicada automaticamente. Na janela Propriedades configuramos a saída do motor "M1" assim como o sentido de rotação "à esquerda" e confirmamos com OK.




- Por baixo deste símbolo colocamos da mesma forma um símbolo do motor e ligamos o motor 2.
- Para aguardar um determinado tempo, utilizamos o elemento Tempo de espera, colocámo-lo por baixo do segundo símbolo do motor e configuramos para 5 segundos.
- A seguir, permitimos ao motor M2 rodar na outra direção (para a direita), aguardamos 2 segundos e desligamos ambos os motores. O nosso programa termina com o símbolo de fim, o semáforo pedestre parado. A figura indica o processo do programa pronto.

Quem não tiver a certeza se está tudo pronto, pode comparar o seu programa com o programa de exemplo fornecido. Para isso, o próprio programa é salvado posteriormente e o arquivo Robô simples 1.rpp é carregado a partir do diretório de exemplo do ROBO Pro (configuração padrão C:\Programas\ROBO Pro\Programas de exemplo\ROBO Mobile Set).

Se tudo estiver em condições, o programa é carregado por download na interface. Após pressionar o botão Download surge uma janela de diálogo. Aí configuramos que o programa deve ser carregado na memória FLASH 1 e iniciado imediatamente após o download.

Logo após o download o nosso modelo se põe em marcha, roda e, a seguir, permanece parado. Se desejarmos reiniciar o programa, premimos a tecla Prog na interface. O LED Prog1 volta a piscar, enquanto o programa corre. A seguir, se acende permanentemente. O programa na memória FLASH da interface permanece parado quando a alimentação de corrente é interrompida na interface.

Experimentamos ao mesmo tempo que retiramos o conector na bateria. Voltamos a encaixar o conector, selecionamos o programa salvado, ao mesmo tempo que premimos a tecla Prog até acender o LED Prog1. Voltamos a premir a tecla e iniciamos o programa.

É muito pouco aquilo que o robô faz até agora, não? Então vamos alargar um pouco a tarefa.

**Tarefa 2 (Level 1):**

**Para que o nosso robô não pare logo após 7 segundos, pretendemos apresentar-lhe agora a dança.**

- **Faça-o andar para em frente, para a esquerda, para a direita, para trás, e a diferentes velocidades.**
- **O decurso se deve repetir até o programa ser concluído com o botão Prog na interface.**

**Dicas:**

- Inverta simplesmente a polaridade dos motores de forma que o robô se desloque nas direções desejadas.
- Pode configurar as velocidades dos motores na janela Propriedades com cada símbolo do motor de 1 a 8. Rodando M1 e M2 com diferentes velocidades na mesma direção, o robô faz uma curva.
- Para que o programa se repita constantemente, arraste a linha de conexão da saída do último elemento do programa para a linha que acompanha o primeiro elemento.
- Encontra um exemplo pronto em Robô simples 2.rpp.

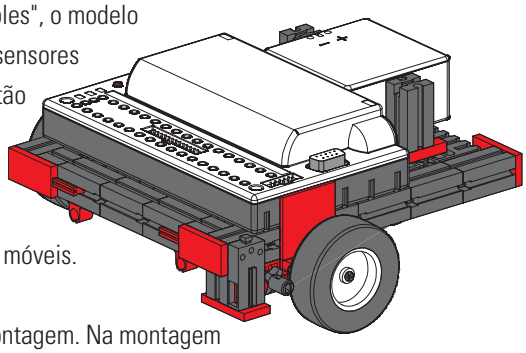
Parabéns, conseguiu construir e programar sozinho o seu primeiro robô. Não é especialmente inteligente, pois não detecta obstáculos e cai da mesa se não prestar atenção. Mas isso ainda se poderá alterar no decorrer de outras experiências.

# Robôs rolantes inteligentes

■ Para que um robô possa detectar o seu ambiente, ele necessita de sensores. Com as seguintes propostas de modelos são apresentadas algumas variantes de robôs móveis, nos quais é experimentada a aplicação de diferentes sensores. Nesse caso, depende de associar não só os estados internos do robô, por exemplo medição da distância através de rodas movidas por impulsos, mas também sinais do exterior, por exemplo na busca de luz ou de trilha. A este modelo são propostas determinadas tarefas. Elas devem servir de estímulo e familiarizá-lo com a matéria. Os programas sobre as tarefas individuais se encontram no diretório ROBO Pro em \Programas de exemplo\ROBO Mobile Set\. Procure encontrar outras funções para os modelos. Ao dar uma olhada nos seguintes exemplos, certamente que vão surgir mais idéias.

## Modelo básico

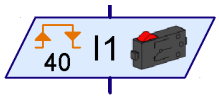
■ Em comparação com os nossos primeiros robôs "robô simples", o modelo básico é mais estável e mais robusto. Contém, além disso, 2 sensores para a medição da distância que são constituídos por um botão e por uma roda movida por impulsos. A roda movida por impulsos está conectada com o eixo do motor e aciona quatro vezes uma tecla em cada rotação do motor.
Este modelo serve como base para outros modelos de robôs móveis.



Construa o modelo básico de acordo com as instruções de montagem. Na montagem proceda com muito cuidado. Quando tudo estiver pronto do ponto de vista mecânico, verifique a mobilidade dos motores, conectando cada motor sem a interface diretamente à bateria.

**Tarefa 1 (Level 1):**

- **Programar a interface de forma que o modelo se desloque 40 impulsos para a frente. Para a medição dos impulsos utilize o botão de contagem na entrada I1.**
- **Meça o percurso que o modelo percorre e calcule que distância é percorrida por impulso.**
- **Repita a tentativa 3 vezes e tome nota na tabela a forma como os valores oscilam.**




**Dicas:**

- A seguir, ligue ambos os motores (sentido de rotação à esquerda).
- Para contar os impulsos em I1, utilize o elemento do programa **Contador de pulsos**.
- Conte ambos os flancos de impulsos (0-1 ao premir, 1-0 ao largar o botão). Pode configurar isto na janela Propriedades em **Tipo de impulso**. Dessa forma, aumenta a exatidão da medição da distância.
- A seguir, volte a desligar os motores e termine o programa.
- Pode consultar o programa já concluído em Modelo básico1.rpp.

**Resultado:**

| | Quantidade de impulsos | Percurso percorrido | Percurso/impulso |
|---|---|---|---|
| **Tentativa 1** | **40** | | |
| **Tentativa 2** | **40** | | |
| **Tentativa 3** | **40** | | |
| | | | |

De forma aproximada se pode anotar que o modelo percorre cerca de uma distância de um centímetro por impulso.

Entretanto, sabe também que sentido de rotação tem de configurar para os motores individuais, para que o modelo se desloque numa determinada direção. Anote estes conhecimentos na seguinte tabela para que não tenha de pensar neles em cada alteração do sentido do deslocamento. Se cablou corretamente os modelos, como está descrito nas instruções de montagem, o sentido de rotação à esquerda significa em cada motor que a roda gira para a frente. É desta forma que são programados os motores em todos os programas de exemplo.

**Complete a tabela:**

| Sentido marcha modelo | Sentido rotação M1 | Sentido rotação M2 |
|---|---|---|
| **Para a frente** | **Esquerda** | **Esquerda** |
| **Para trás** | | |
| **Esquerda** | | |
| **Direita** | | |
| **Pare** | | |

Para não precisar colocar dois símbolos do motor na tela em cada mudança de direção, pode criar para cada direção um subprograma que assume esta tarefa. Isto facilita muito a programação. A forma como criar subprogramas está descrita no manual de instruções do software ROBO Pro no capítulo 4. Logo que tenha lido este capítulo, pode se aventurar na próxima tarefa. No ROBO Pro passa para o **Level 2**.

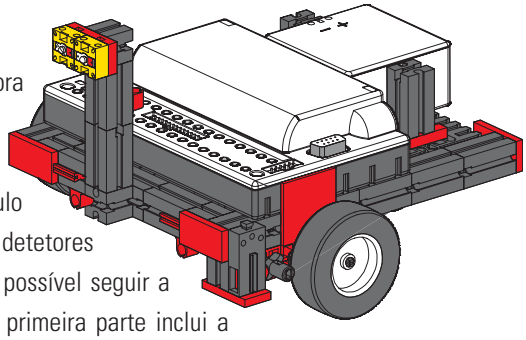
**Tarefa 2 (Level 2):**

- **Crie para cada sentido de marcha um subprograma.**
- **Programe o modelo de forma que percorra um quadrado com bordas de um metro.**
- **Qual é a precisão da repetição?**

**Dicas:**

- Crie, em primeiro lugar, um subprograma **Para a frente**. Pode criar os outros subprogramas através da cópia deste subprograma. Tem de adaptar apenas os sentidos de rotação dos motores.
- Para virar para a esquerda e para a direita, utilize uma velocidade reduzida. Isso aumenta a precisão.
- Para a contagem de impulsos utilize novamente o elemento Contador de pulsos e o botão na entrada I1.
- Carregue o programa para experimentar primeiro na RAM, até descobrir quantos impulsos necessita para efetuar uma rotação de 90°. Em primeiro lugar, o carregamento na RAM é mais rápido do que o carregamento na memória FLASH e, em segundo lugar, a memória Flash tem apenas uma vida útil "limitada" de aprox. 100.000 downloads.
- O programa já concluído se chama Modelo básico2.rpp.

## O robô que segue um foco de luz



■ Depois de ter analisado o modelo básico, o robô deve agora aprender a reagir a sinais do ambiente. Semelhante à mariposa da nossa experiência de pensamento do primeiro capítulo, ele deve detectar a fonte de luz e segui-la. O módulo contém 2 fototransístores que estabelecemos como sendo detetores de luz. Cada sensor reage a um motor de forma que seja possível seguir a fonte de luz. O programa é composto por duas partes. A primeira parte inclui a procura de uma fonte de luz e na outra parte é realizado o seguimento ou a ativação da fonte de luz. Para isso são utilizados outros subprogramas. Após a conexão é ativado o subprograma Busca de luz. Este subprograma só é concluído depois de ser encontrada uma fonte de luz. O programa principal tenta dirigir o robô na direção da fonte de luz. Sempre que a direção do robô se desvia muito da linha ideal, um dos sensores não é iluminado pela fonte de luz. Por conseguinte, o robô corrige o seu sentido de marcha de forma que ambos os sensores possam detectar a fonte de luz.

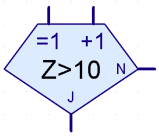
Primeiro, monte o modelo que segue um foco de luz conforme descrito nas instruções de montagem.

**Tarefa 1 (Level 2):**

- **Programe, em primeiro lugar, a função "Busca de luz". O robô deve virar lentamente e pelo menos a 360°. Se, durante a busca, for encontrada uma luz, o robô pára. Caso contrário, ele gira 360° na outra direção. Se ele não encontrar qualquer fonte de luz, ele deve aguardar 5 segundos antes de voltar a iniciar a busca.**
- **Se a busca de luz for bem sucedida, o modelo deve ativar a fonte de luz. Se a fonte e luz se movimentar para a esquerda ou para a direita, o robô deve seguir os movimentos da luz. Se ele perder o contato, o programa deve iniciar novamente a busca de luz. Veja se consegue atrair o robô com uma lanterna de bolso e conduzi-lo por um percurso com obstáculos.**

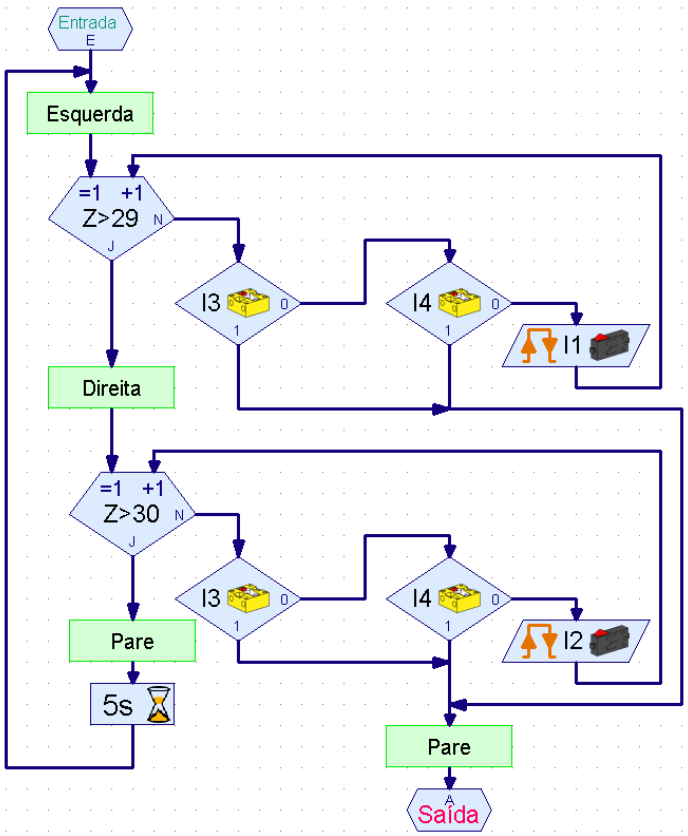
**Dicas:**

- Para os diferentes sentidos de marcha utilize os subprogramas que já programou para o modelo básico. Assim que o programa Modelo básico2.rpp esteja aberto, encontra **na janela grupos de elementos** do ROBO Pro em **Programas carregados** o programa Modelo básico2 e por baixo os subprogramas que estão incluídos no programa Modelo básico2. Pode introduzir facilmente estes subprogramas no novo programa.
- Para o subprograma "Busca de luz" utilize o elemento **Ciclo de contagem**. (sobre a descrição do elemento ver manual de instruções ROBO Pro).




**Ciclo de contagem**

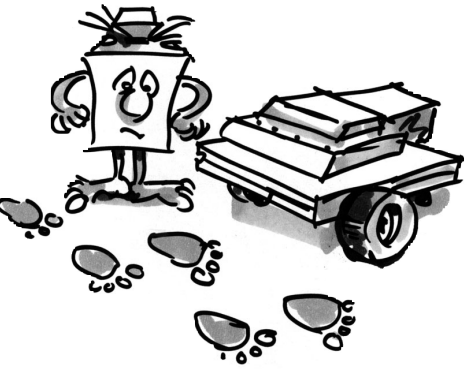
- No ciclo entre a ligação "N" e a ligação "+1" consulte os fototransístores e conte um impulso no botão de pulso I1. O ciclo é percorrido até o robô encontrar luz ou ter realizado uma rotação de 360°. Experimente com que freqüência ele tem de percorrer o ciclo até fazer uma rotação completa e coloque, em função disso, o valor "Z" no elemento Ciclo de contagem.
- Programe um segundo ciclo semelhante para a busca na direção inversa.
- Se o robô encontrar luz, ele pára e abandona o subprograma.
- Aqui está representado o subprograma completo:



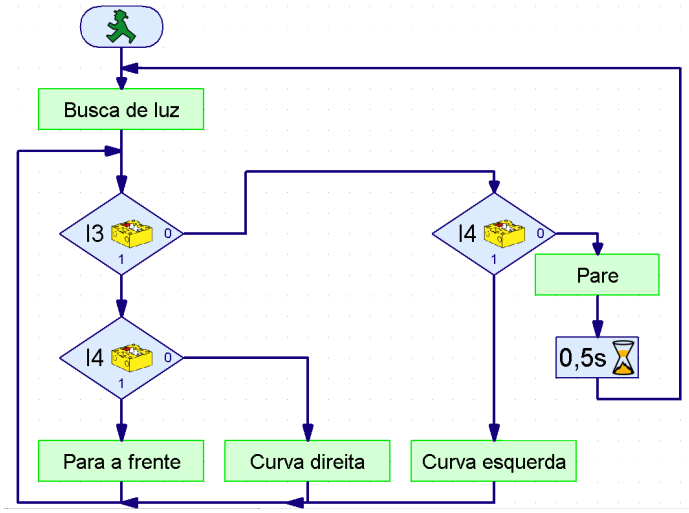

- No programa principal volte a consultar os fototransístores e comande os motores em função de que fototransístor detecta luz:

| | |
|---|---|
| **Luz em I3 e I4** | **Para a frente** |
| **Luz apenas em I3** | **Curva à direita** |
| **Luz apenas em I4** | **Curva à esquerda** |
| **Luz não detectada** | **Pára, volta para o subprograma Busca de luz** |

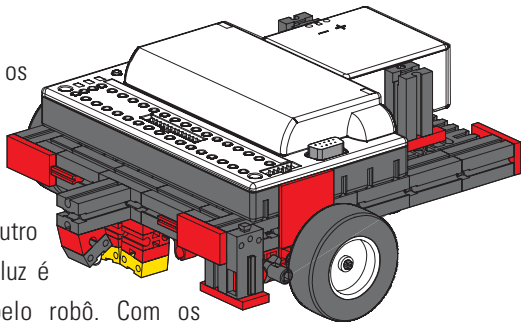
- Crie a curva à direita e à esquerda por meio de diferentes velocidades do M1 e M2 no mesmo sentido de rotação. Com isso se cria um estilo de marcha muito harmonioso.



- O programa principal tem o seguinte aspecto:
- Pode consultar o programa já concluído em Busca de luz.rpp.
- Utilize uma lanterna de bolsa como fonte de luz. Tente focalizar o feixe de luz com intensidade para que ambos os fotosensores sejam iluminados pela fonte de luz. Preste atenção para o fato de em lugares muito iluminados a sua lanterna de bolso poder ser ofuscada por outras fontes de luz, como por exemplo luz solar de uma grande janela. O robô passa ao lado da sua lâmpada em direção à luz mais forte.




## O robô que segue uma trilha



■ A busca e seguimento são propriedades essenciais que os seres inteligentes possuem. Com o robô que segue um foco de luz construiu e programou um robô que reagiu a sinais diretos do seu objetivo.

Com o robô que segue uma trilha empregamos um outro princípio de busca. Em vez da marcha para a fonte de luz é marcada uma linha preta que deve ser seguida pelo robô. Com os fototransístores esta tarefa é relativamente fácil de efetuar. A luz da marcação refletida é medida e os motores são corrigidos. Para que funcione corretamente, a linha é iluminada com a lâmpada. Preste atenção para o fato de os fotosensores não serem ofuscados pela luz difusa da lâmpada através de uma disposição desfavorável. Neste contexto a luz da lente óptica da lâmpada incandescente atua de forma especialmente vantajosa.

Construa agora o modelo que segue uma trilha de acordo com as instruções de montagem.

**Tarefa 1:**

- **Em primeiro lugar, escreva um subprograma com o qual a trilha é procurada. O robô deve rodar uma vez em circuito.**
- **Se não encontrar qualquer trilha, deve se deslocar um pouco para a frente e voltar a procurar em seguida. Para a detecção de trilhas são consultados os fototransístores.**
- **Quando o robô descobrir a trilha, deve segui-la.**
- **Se a trilha terminar ou se o robô a perder, devido, por exemplo, a uma alteração brusca da direção da trilha, a procura deve ser reiniciada.**

**Dicas:**

- Após ligar a lâmpada tem de se aguardar um instante (aprox. um segundo) antes dos fototransístores serem consultados. Caso contrário, o fototransístor detecta "escuro", ou seja uma trilha, onde não há nenhuma, porque a consulta foi efetuada antes da lâmpada estar corretamente acesa.
- Você pode utilizar aprox. 20mm de fita isolante preta larga como trilha ou pode pintar uma trilha preta dessa largura com caneta de feltro num papel branco. As curvas não devem ser muito acentuadas, caso contrário o robô perde a trilha demasiadas vezes.
- Primeiro, com o teste de interface verifique se a sua trilha foi detectada corretamente pelos fototransístores. Não se esqueça de desligar a lâmpada.
- Ajuste a lâmpada de modo que os dois fototransístores forneçam o valor 1 relativamente ao fundo claro, mesmo quando os motores M1 e M2 são ligados. Se a sua bateria estiver um pouco fraca, ao ligar os motores a lâmpada fica um pouco escura. Caso esta não esteja corretamente ajustada, um fototransístor pode indicar "escuro" apesar de não ter encontrado qualquer trilha.
- A busca de trilha funciona do mesmo modo que a busca de luz. Você apenas tem de adaptar a busca de modo que, em caso de busca malograda, o modelo se desloque um pouco em linha depois de uma rotação completa, antes de continuar a busca.
- Preste atenção para que o modelo tenha de se deslocar em linha reta, em caso de seguimento de trilha, quando os dois fototransístores fornecerem o valor "escuro" (=0).
- Pode consultar o programa já concluído em Busca de trilha.rpp.

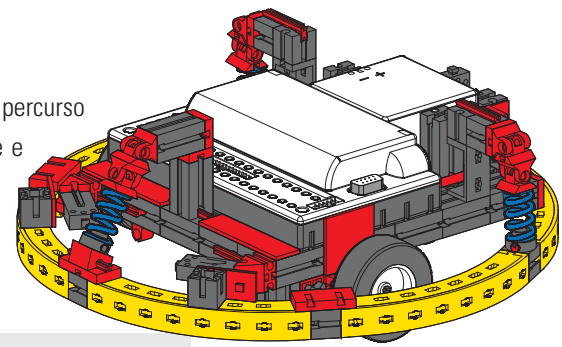
**Tarefa 2:**

- **Crie uma trilha com várias curvas acentuadas. Qual é o raio que o modelo ainda consegue efetuar?**
- **Ao corrigir a trilha experimente as diferentes velocidades dos M1 e M2. Que combinação proporciona o melhor resultado?**
- **Crie uma trilha com um percurso circular. Procure otimizar as velocidades de modo que o robô consiga efetuar o percurso no menor tempo possível. Esta tarefa é mais adequada para uma competição com mais robôs.**

## Robô com detecção de obstáculos

■ Todos os robôs construídos até agora conseguem percorrer uma determinada distância assim como seguir uma fonte de luz ou uma trilha. Mas o que sucede quando se coloca um obstáculo no caminho? Ou o obstáculo é empurrado para o lado ou o robô se desloca contra ele até que a bateria fique vazia. Naturalmente seria muito mais inteligente se o robô detectasse o obstáculo e se desviasse respectivamente. Para isso, o robô contém um pára-choques móvel em volta com três botões. Com este pára-choques ele pode distinguir se existe um obstáculo à esquerda, direita ou atrás.
O modo como ele deve reagir é apenas uma questão de programação.

Primeiro construa o modelo "robô com detecção de obstáculos". Para a medição do percurso é necessário apenas um botão (I1). Por isso, o botão I2 é retirado do modelo base e utilizado para a detecção de obstáculos.



**Tarefa 1 (Level 2):**

- **Primeiro, o robô deve se deslocar em linha reta. Caso ele bata num obstáculo pela esquerda (I4), ele deve recuar um pouco e, a seguir, desviar para a direita.**
- **Caso ele bata num obstáculo pela direita (I3), ele deve recuar um pouco e, a seguir, desviar para a esquerda.**

**Dicas:**

- A detecção de obstáculos ao fazer marcha a ré ainda não é tida em conta.
- Os botões são consultados no programa principal. Dependendo de qual botão é acionado, o modelo se desvia para a esquerda ou para a direita. Isto se efetua num subprograma.
- O número de pulsos na rotação à direita se devia diferenciar do número de pulsos na rotação à esquerda (p. ex. 3 pulsos para a direita, 5 pulsos para a esquerda). Caso contrário, o modelo pode ir para um canto e não conseguir descobrir onde está, porque gira sempre a mesma distância para a direita e para a esquerda.
- O programa já concluído se chama Obstáculo1.rpp.

O detetor de obstáculos ainda não consegue fazer duas coisas: Ao fazer marcha a ré não detecta quaisquer obstáculos. Ele também ainda não reconhece quando se encontra um obstáculo diretamente à sua frente. Mas, ele poderia detectar ambos. Se durante a marcha a ré I5 for premido, se encontra um obstáculo atrás do modelo. Se na marcha à frente forem premidos simultaneamente I3 e I4, então se encontra um obstáculo diretamente à frente do modelo. Neste caso, o robô poderia girar em 90°. No total, estão disponíveis as seguintes possibilidades, às quais o robô deve reagir:

| Obstáculo | Botão | Reação |
|---|---|---|
| à direita | apenas I3 | Desviar para a esquerda (rotação de aprox. 30°) |
| à esquerda | apenas I4 | Desviar para a direita (aprox. 45°) |
| à frente | I3 e I4 | Desviar para a esquerda (aprox. 90°) |
| atrás | I5 | É apenas consultado na marcha à ré. Parar e, a seguir, continuar a desviar conforme planejado |

Para solucionar esta tarefa facilmente, alguns novos elementos de programa, como p. ex. operadores (p. ex. E, OU), do ROBO Pro Level 3 podem ser uma boa ajuda. No Level 3 também é possível trocar dados entre diferentes elementos através das setas cor de laranja. Para isso, mude para este level no software. A seguir, você deve pegar no manual de instruções ROBO Pro e ler o capítulo 5 cuidadosamente. A seguir, você está preparado para a próxima tarefa.

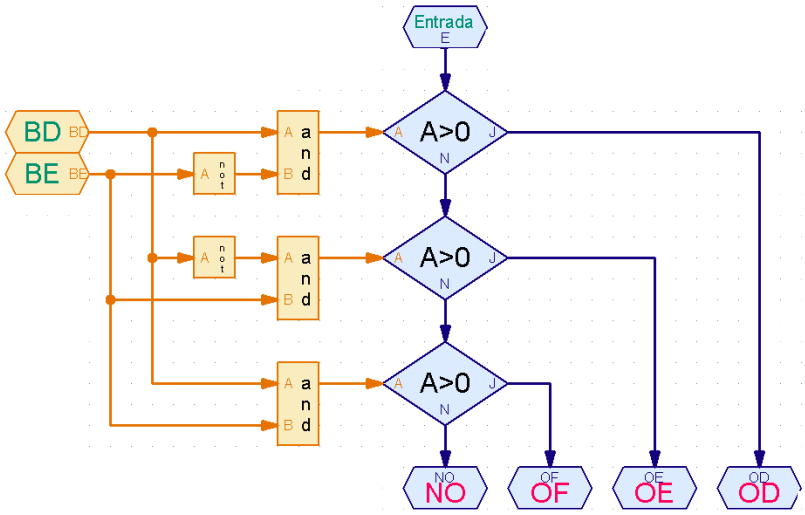
**Tarefa 2 (Level 3):**

- **Remodele o seu programa de obstáculos de modo que o modelo reaja de acordo com a tabela acima.**
- **Para isso, utilize as possibilidades do ROBO Pro level 3.**

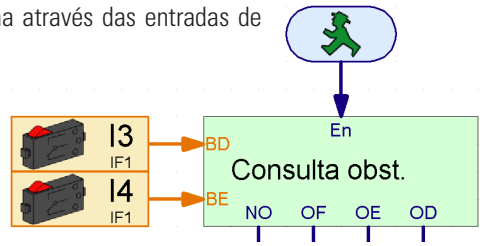
**Dicas:**

- Com a ajuda dos operadores, as diferentes combinações possíveis de teclas são consultadas num subprograma "Consulta de obstáculos". O subprograma possui uma saída própria para cada possibilidade.

Entrada de dados BD = Botão direito
Entrada de dados BE = Botão esquerdo
Saída NO = Nenhum obstáculo
Saída OF = Obstáculo à frente
Saída OE = Obstáculo à esquerda
Saída OD = Obstáculo à direita

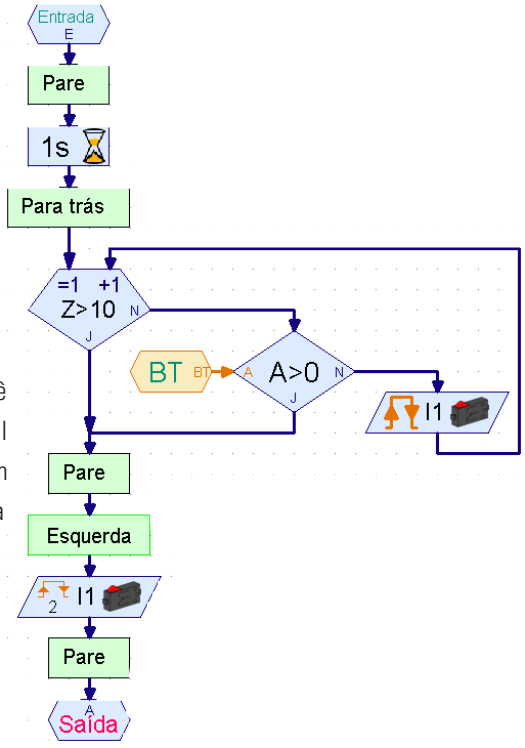



- Para que se detecte de imediato quais botões são consultados, você deve colocar os elementos cor de laranja dos botões no programa principal e os ligar ao subprograma através das entradas de dados.
- Nos vários subprogramas de desvio, I5 é consultado durante a marcha a ré. O modelo se desloca de marcha a ré até que o número de pulsos ajustado seja alcançado ou que I5 seja pressionado. I5 é novamente colocado no programa principal, para que seja visível de imediato em que subprogamas ele é consultado.




- Pode consultar o programa completo em Obstáculo2.rpp

Uma vantagem da técnica de programação utilizada nesta tarefa é que você pode ver diretamente no programa principal que botão é consultado em qual subprograma. Caso pretenda alterar a entrada, você tem de fazer isso num local e não procurar em todos os subprogramas o local onde o botão poderia estar escondido. Além disso, se pode estabelecer uma combinação lógica muito evidente com os operadores. Em principio, isto também pode ser efetuado com elementos de ramificação, mas rapidamente se torna de difícil compreensão quando são consultados vários casos.



## Robô que segue um foco de luz com detecção de obstáculos

■ Ainda falta muito para acabarem as possibilidades que o ROBO Mobile Set oferece.
Por isso, agora devem ser combinadas as duas funções: busca de luz e detecção de obstáculos. Do ponto de vista científico o robô está equipado com dois tipos de comportamento. Mas, visto que, os dois modelos de comportamento não podem estar ativos simultaneamente, eles possuem diferentes prioridades. Normalmente o robô responde à busca de luz. Ele detecta um obstáculo, ou seja um perigo para ele, e o comportamento para evitar o obstáculo fica ativo. Caso tudo se encontre na área verde, o robô pode continuar a procurar a fonte de luz.

Se engenheiros profissionais de software se ocuparem com uma tarefa tão exigente, eles não começam a programar à sorte, mas sim utilizam uma determinada estratégia para o desenvolvimento do programa. Um destes métodos é chamado "Rascunho Top-Down". Neste procedimento, todo o sistema é definido a partir do topo, sem se preocupar com todos os detalhes logo no início. Nós também utilizamos este método neste problema.

**Tarefa 1 (Level 3):**

**Programe o robô com o seguinte tipo de comportamento:**

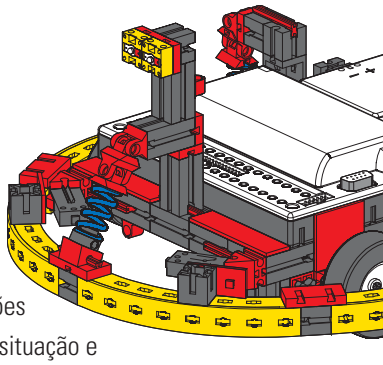
- **Busca de uma fonte de luz.**
- **Logo que a tenha encontrado, siga-a.**
- **Pelo caminho surge um obstáculo, evite-o.**
- **A seguir, volta a buscar uma fonte de luz**

**Para a solução, utilize os elementos do programa do ROBO Pro Level 3.**
**Solucione esta tarefa "a partir de cima" de acordo com o procedimento Top-Down.**

**Dicas:**

Primeiro você divide a tarefa em três partes:

- Consulta se o robô detecta uma fonte de luz (subprograma "Luz")
- Consulta se ele bateu num obstáculo (subprograma "Obstáculo")
- Dependendo destes resultados, você comunica ao robô o que ele deve fazer (subprograma "Deslocamento")



Agora pondere para os subprogramas "Luz" e "Obstáculo", quais situações diferentes o robô pode percepcionar. Você designa um valor numérico a cada situação e o salva na variante com a ajuda de um elemento de comando. De cada situação resulta uma reação, que é executada no subprograma "Deslocamento".

**Subprograma Luz:**

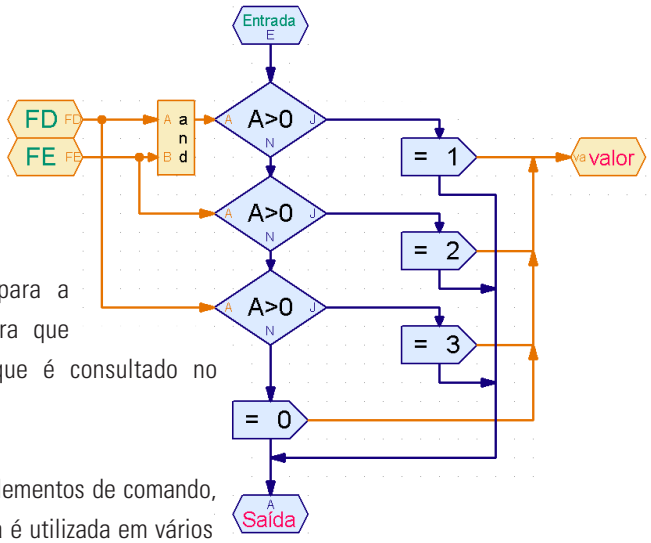
| N.º | Situação | Estado dos sensores | Reação |
|---|---|---|---|
| 0 | Nenhuma fonte de luz disponível | I6=0; I7=0 | Buscar luz |
| 1 | Fonte de luz à frente do robô | I6=1; I7=1 | Deslocar em linha reta |
| 2 | Fonte de luz à esquerda do robô | I7=1 | Efetuar curva à esquerda |
| 3 | Fonte de luz à direita do robô | I6=1 | Efetuar curva à direita |

**Subprograma Obstáculo:**

| N.º | Situação | Estado dos sensores | Reação |
|---|---|---|---|
| 4 | Obstáculo diretamente à frente do robô | I3=1; I4=1 | Desviar 90° |
| 5 | Obstáculo à direita do robô | I3=1 | Desviar para a esquerda |
| 6 | Obstáculo à esquerda do robô | I4=1 | Desviar para a direita |

Agora você ainda tem de apresentar estas detecções no ROBO Pro com elementos do programa.

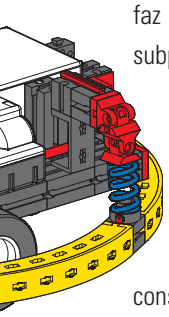
**Subprograma Luz:**




FD=Fototransístor direito
FE=Fototransístor esquerdo

Volte a colocar os elementos para a consulta dos fototransístores, para que possa detectar de imediato o que é consultado no subprograma.

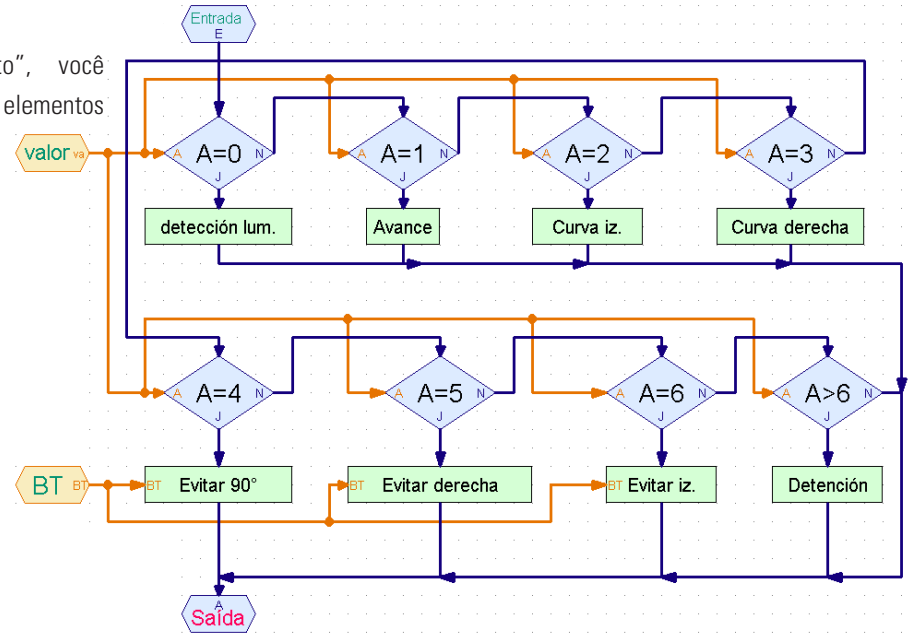
A variante, que salva o valor dos elementos de comando, faz parte do programa principal. Ela é utilizada em vários subprogramas. Ela é conectada ao subprograma através de uma saída de dados.



Você gera o subprograma "Obstáculos" de acordo com o mesmo princípio com o subprograma "Luz".

No subprograma "Deslocamento", você consulta o valor atual da variante com os elementos de ramificação e programa a respetiva reação:

BT=Botão traseiro



Por último, tem de gerar os subprogramas, que são utilizados neste subprograma.
Mas, espere um pouco! Já estão quase todos gerados. O subprograma Busca de luz, p. ex., pode ser copiado do programa para o modelo que segue um foco de luz. Se não souber como isso se efetua, leia o capítulo 4 do manual de instruções ROBO Pro.

**Mas cuidado:**
No modelo que segue um foco de luz, os fototransístores estavam conectados na entrada I3 e I4. Mas agora estão na I6 e I7. Além disso, o botão I1 é consultado para contar os pulsos quando gira para a esquerda e o botão I2 quando gira para a direita. Agora existe apenas o I1 para contar os pulsos, o que normalmente também funciona bem. Você tem de adaptar o subprograma busca de luz depois de o copiar. Visto que a consulta dos botões está escondida no subprograma, esta passa despercebida com facilidade. Isto não acontece, se você colocar as entradas no programa principal e as conectar com o subprograma através das entradas de dados. Mas esta possibilidade não é possível com o robô que segue um foco de luz.

Os subprogramas para desvio também já estão disponíveis, nomeadamente no modelo com detecção de obstáculos. Neste caso o botão I5, que é consultado adicionalmente ao fazer marcha a ré, está saliente.

Você pode ver o programa concluído em Luz do obstáculo.rpp.
À primeira vista, o programa principal parece muito simples e fácil de compreender. E, no entanto, há muito por detrás dos subprogramas. Mas com a ajuda do procedimento passo-a-passo com o método Top-Down, você pode solucionar um problema com esta complexidade.

Além disso, se um dos seus amigos possuir um módulo ROBO Mobile Set, vocês podem continuar as experiências com este robô. Em cada um dos robôs é montada apenas uma fonte de luz. A seguir, os robôs se buscam mutuamente.

## Robô com detecção de bordas

■ Depois de você ter visto no último exemplo como se procede na programação de um problema mais complexo, agora você pode se dedicar a outro comportamento muito importante de um robô móvel. Ele deve aprender a não cair de uma mesa. Se o robô bater num obstáculo, na maioria dos casos, isso não tem qualquer importância. Mas se ele cair de uma mesa com um metro de altura, ele pode sofrer alguns danos, apesar dos módulos da fischertechnik serem muito estáveis. Por esta razão, o robô recebe sensores com os quais ele pode detectar as bordas. Este detetores de bordas são compostos por um botão, que é ativado por uma roda rotativa apoiada. Esta roda pode se movimentar para os dois lados. Logo que a roda ultrapasse a borda da mesa, esta cai, o botão não é mais ativado, o programa detecta que o modelo se encontra num precipício e reage de forma correspondente. O robô tem um total de 4 detetores de bordas de modo que este pode procurar precipícios na marcha à frente e na marcha à ré de ambos os lados. Para isso, este modelo não tem qualquer botão de pulso para a medição do percurso. O caminho percorrido é controlado pela duração da conexão dos motores.

Primeiro, monte o modelo conforme descrito nas instruções de montagem.

Verifique com exatidão se os detetores de bordas são corretamente ativados:

- quando o modelo chega à borda da mesa e o botão for novamente premido exatamente
- quando a roda volta para cima da mesa.

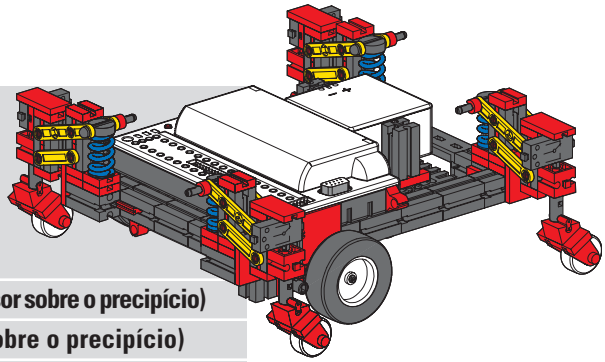
Eventualmente, você tem de ajustar um ou outro botão um pouco para cima ou para baixo.

**Tarefa 1 (Level 3):**

- **Primeiro, pondere sobre o modo como o robô deve reagir, quando ele chega a um precipício.**
- **Se pensar cuidadosamente, lhe irá ocorrer que existem muitas possibilidades de combinação relativamente a quais sensores se podem encontrar no precipício. A ativação pode ocorrer em 1, em 2 ou 3 sensores diferentes simultaneamente ou até em 4 sensores.**
- **Como é que o robô deve reagir em cada uma das vezes?**

**Dicas:**

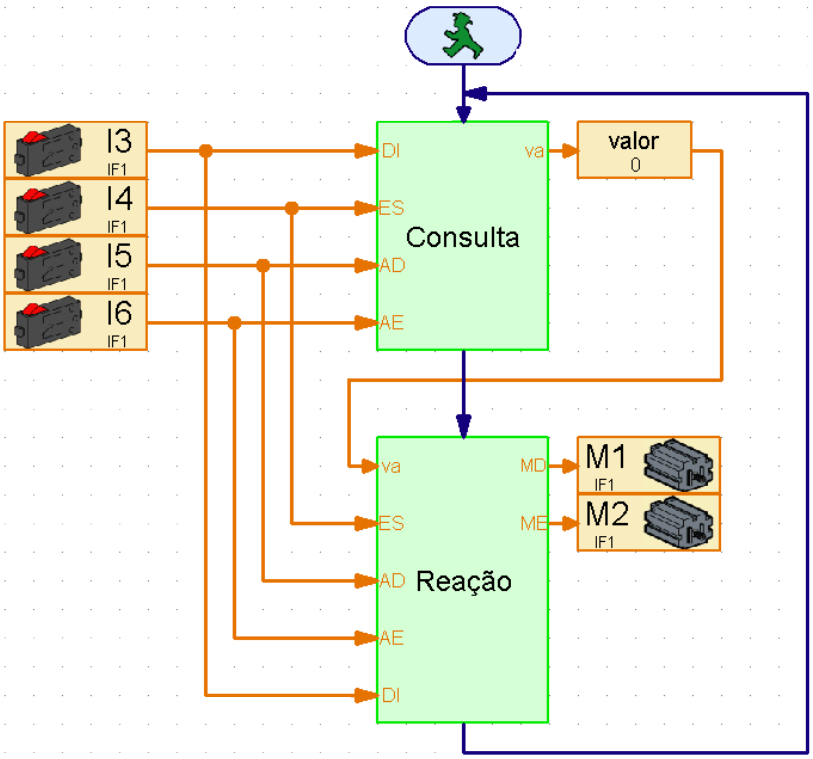
Pode encontrar a solução na tabela seguinte. Os sensores, que se encontrarem sobre o precipício (botão=0) estão marcados com uma cruz. Cada combinação contém um número. No programa, que é gerado mais tarde, é atribuído o número respetivo a cada possibilidade. De acordo com o número, o robô reage à situação atual. Mas isso será mais tarde. Primeiro, reflita apenas sobre o lugar onde o robô deve estar para que a combinação se efetue e verifique se ele reage.



| N.º | Frente direita (I3) | Frente esquerda (I4) | Atrás direita (I5) | Atrás esquerda (I6) | Reação |
|---|---|---|---|---|---|
| 0 | | | | | Para a frente (Nenhum sensor sobre o precipício) |
| 1 | ● | ● | ● | ● | Parada (os 4 sensores sobre o precipício) |
| 2 | ● | ● | ● | | Girar um pouco para a direita |
| 3 | ● | ● | | ● | Girar um pouco para a esquerda |
| 4 | ● | | ● | ● | Girar um pouco para a esquerda |
| 5 | | ● | ● | ● | Girar um pouco para a direita |
| 6 | ● | ● | | | Primeiro para trás, depois gira para a direita |
| 7 | ● | | ● | | Girar um pouco para a esquerda |
| 8 | ● | | | ● | Girar um pouco para a esquerda |
| 9 | | ● | ● | | Girar um pouco para a direita |
| 10 | | ● | | ● | Girar um pouco para a direita |
| 11 | | | ● | ● | Se desloca um pouco para a frente |
| 12 | ● | | | | Primeiro para trás, depois gira para a esquerda |
| 13 | | ● | | | Primeiro para trás, depois gira para a direita |
| 14 | | | ● | | Se desloca um pouco para a frente |
| 15 | | | | ● | Se desloca um pouco para a frente |

Bastante difícil, não? Mas, não se preocupe, existe um programa já concluído para este modelo, que utiliza todas as vantagens do ROBO Pro. É o Cantos.rpp.

Os elementos mais importantes se encontram no programa principal de modo que você possa compreender todo o decurso. A consulta complexa do botão e a ativação dos motores estão escondidas nos subprogramas. Primeiro o programa principal:




O decurso se inicia com a consulta dos 4 botões. À esquerda você pode ver qual botão é consultado. Eles estão conectados ao subprograma através das entradas de dados. O subprograma "Consulta" determina qual botão é pressionado e emite o valor descrito na tabela. Este valor é atribuído às variantes com o mesmo nome, que você pode reconhecer no programa principal. O valor da variante é transmitido ao subprograma "Reação", que, a seguir, ativa um dos dois motores dependendo deste valor. No subprograma "Reação" também são lidos os botões, visto que os sensores de bordas são consultados, enquanto o modelo se desvia.

Agora você pode alterar a ocupação dos botões na interface, assim como as saídas do motor, sem ter de examinar exaustivamente em todos os subprogramas o local onde poderia estar escondido um elemento de entrada ou um símbolo do motor. Cada entrada e cada saída só surge uma vez.

Você pode utilizar esta técnica de programação principalmente quando um subprograma tiver de ser utilizado em muitos modelos e você antes não souber exatamente que entradas e saídas da interface devem ser utilizadas.

Se você tiver ficado curioso, pode dar uma olhada nos subprogramas e tente compreender. O princípio da programação é semelhante ao do modelo "robô que segue um foco de luz com detecção de obstáculos"

**Tarefa 2 (Level 3):**

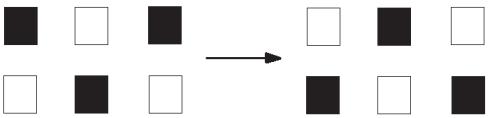
**Carregue o programa na interface e coloque o modelo a se deslocar numa mesa.**

- **O modelo reage sempre de modo correto?**
- **Ele deveria responder de outro modo a determinadas combinações de botões?**
- **Se necessário, otimize o programa.**

# O robô andante

■ Depois de nos termos ocupado detalhadamente com os robôs rolantes, debrucemo-nos agora sobre um outro tipo de movimentação, que podemos utilizar para robôs móveis, o andar.
A forma de caminhar dos insetos se apropria de modo excelente como modelo para o acionamento de "objetos com seis pernas mecânicas". Na chamada "marcha com três pés" se elevam simultaneamente do chão sempre três das seis pernas, a perna dianteira e traseira de um dos lados, juntamente com a perna central do outro lado:

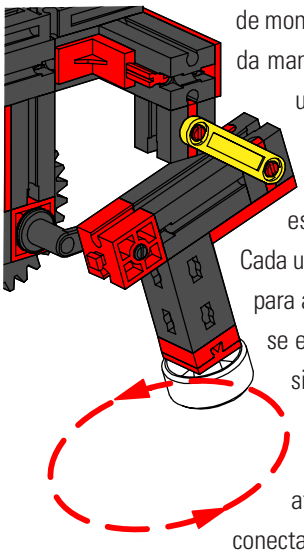
**Marcha com três pés**



As pernas que permanecem no chão (apresentadas a preto), formam um objeto de três pernas estável, de modo a que o modelo se mantenha sempre estável e não vire ao andar.

As pernas do robô andante da fischertechnik são construídas de tal modo, que resultam numa articulação de quatro membros. A forma de construção dos quatro membros aqui utilizada é denominada de "Mecanismo corrediça-manivela oscilante". Acionados por uma manivela, os membros munidos de movimento da articulação efetuam movimentos oscilantes. As distâncias entre os membros individuais e o local da extremidade do pé (isso é a extremidade inferior do pé) estão escolhidas de tal forma, que a extremidade do pé descreve um movimento elíptico, quando a manivela de acionamento roda. Através disso é originado um movimento, que se assemelha a um passo ao caminhar.

As 6 manivelas, que acionam as pernas têm de ser ajustadas exatamente como indicado nas instruções de montagem. As três pernas que assentam simultaneamente no chão, têm a mesma posição da manivela. As manivelas das 3 pernas, que neste momento se mantêm elevadas, sofrem uma rotação de 180°. A posição correta das manivelas entre si garante que o modelo se possa deslocar numa seqüência de passos correta, a "marcha com três pés".



As porcas de cubo, com as quais as rodas dentadas são fixadas nos eixos, têm de estar bem apertadas, de modo a que as manivelas não saiam do lugar durante o andar. Cada um dos lados direito e esquerdo do modelo são acionados por um motor (isso é necessário para andar em curvas). Por isso deve-se ter em atenção, que a perna central de um dos lados se encontre sempre na mesma posição que ambas as pernas exteriores do outro lado. Esta sincronização é efetuada através de controle por software, através dos botões I1 e I2.

Primeiro, monte o modelo conforme descrito nas instruções de montagem. Controle, através do teste de interface, se todos os botões e motores se encontram corretamente conectados. Sentido de rotação dos motores: Sentido de rotação para esquerda=para a frente.

**Tarefa 1 (Level 1):**
**Programe o andar do robô:**

- **Programe o modelo de forma que este se desloque na marcha com três pés, para a frente.**
- **Utilize os botões I1 e I2 para a sincronização das pernas esquerdas e direitas.**
- **Tenha em atenção que ambas as pernas exteriores de um dos lados e a perna central do lado oposto têm de ter sempre a mesma posição.**

**Dicas:**

- Coloque primeiro as pernas dos lados esquerdo e direito na sua posição de saída. Ligue ambos os motores (sentido de rotação à esquerda).
- O processo só deve continuar quando ambos os botões I1 e I2 não estão premidos (esta consulta é necessária, desde que o modelo tenha de fazer o segundo passo).
- Deixe os motores trabalhar, até o respectivo botão (I1 para M1, I2 para M2) estar novamente premido. Aí é importante que o modelo só inicie o próximo passo quando ambos os botões estão premidos. Aí as pernas estão numa posição correta entre si. A condição para tal é também que as manivelas que acionam as pernas estejam ajustadas corretamente como indicado nas instruções de montagem.
- Agora o processo pode começar pela frente e o robô dá o segundo passo. Agora o modelo anda para a frente, até que pare o programa.
- Pode consultar o programa já concluído em Robô andante1.rpp.

Pode agora, de forma semelhante ao modelo básico rolante, deixar o modelo andar para a esquerda, direita ou para trás, alterando os sentidos de rotação do motor. Para contar os passos, pode utilizar I1 ou I2.

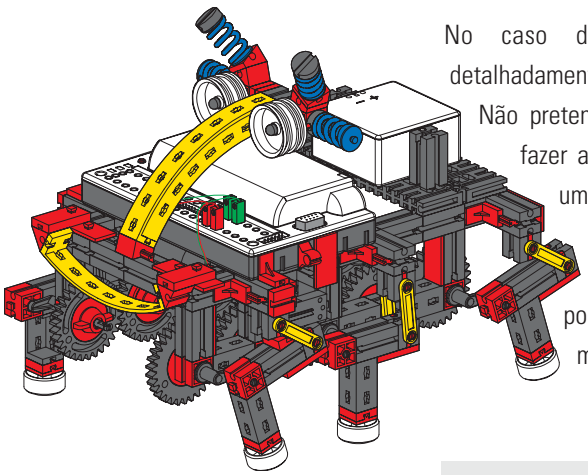
**Tarefa 2 (Level 2):**

- **Programe o seu modelo de forma que este ande 10 passos para a frente, 3 passos para a esquerda, 3 passos para a direita e novamente 10 passos para trás.**
- **Aplique para cada sentido um subprograma próprio.**
- **Para a contagem dos passos, utilize o elemento Ciclo de contagem.**

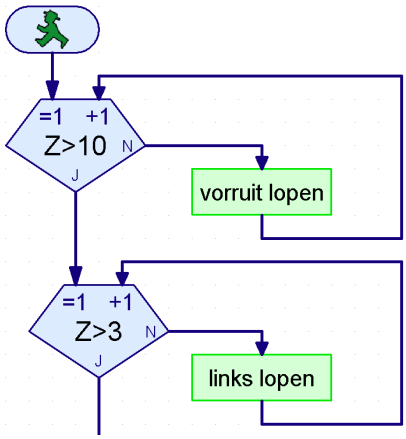
**Dicas:**

- Copie simplesmente o programa Robô andante1.rpp para um subprograma.
- Copie este subprograma as vezes necessárias para os diferentes sentidos de marcha. Altere em cada subprograma os sentidos de rotação de forma que o modelo se movimente no sentido pretendido.
- Utilize o elemento Ciclo de contagem para contar o número de passos para cada sentido de rotação. O modelo dá um passo em cada execução de um subprograma. Se o programa executa o ciclo com o subprograma 10 vezes, o modelo dá 10 passos.

Desta forma, pode programar o seu robô andante com uma seqüência de passos desejada (robô andante 2.rpp).



No caso dos robôs rolantes, abordamos detalhadamente o tema detecção de obstáculos. Não pretendemos nos repetir aqui. Mas tente fazer a transferência deste comportamento uma vez para os robôs andantes. Os sensores para isso estão incluídos no módulo. Durante a programação pode tomar os robôs rolantes como modelo. Bom trabalho!

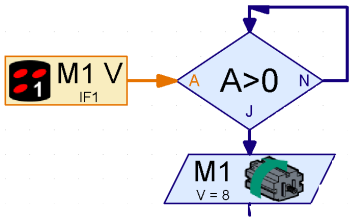


# Possibilidades de expansão

■ O ROBO Interface oferece mais funcionalidade do que a apresentada até ao momento pelos robôs móveis. Para isso são necessários, no entanto, componentes adicionais, que não se encontram incluídos no volume de fornecimento do módulo. Uma vez que estes são também extremamente interessantes para os robôs, gostaríamos de apresentar aqui alguns deles.

## Emissor manual de infravermelhos

■ No ROBO Interface está incluído um diodo receptor de infra-vermelhos do IR Control Set, n.º de art. 30344, para o emissor manual. No software ROBO Pro pode consultar, desse modo, as teclas do emissor manual, como entradas digitais e, assim, p. ex. ligar e desligar motores.




Como exemplo do programa programamos um comando à distância para robô andante. Com as 4 teclas de setas ovais no controle remoto pode comandar o modelo para a frente. Para trás, para a esquerda e para a direita. Só necessita de carregar previamente o programa Robô andante-IR.rpp na interface.

Um outro programa genial em conjunto com o controle remoto é o programa Mobile-Teach-IR.rpp. Através deste programa Teach-In pode comandar à distância um dos robôs rolantes, p. ex. o robô simples ou o modelo básico. O modelo memoriza o percurso percorrido e é capaz de o repetir mais tarde, sempre que desejado. O percurso memorizado é contudo apagado, quando o programa para.
Um programa destes se torna possível através do elemento do programa "Liste" no ROBO Pro. Neste elemento, é possível memorizar e voltar a ler vários valores (ver também o manual de instruções do ROBO Pro). O próprio programa é extremamente complexo, mas a utilização é bastante fácil:

1. Carregar e iniciar o programa Mobile-Teach-In.rpp na memória Flash da interface ROBO.
2. Premir a tecla **M1** ▸/▸▸ no controle remoto. O "processo de programação" é iniciado.
3. Comandar o modelo na direção pretendida, através das teclas de setas ovais.
4. Premir a tecla **M2** ▸/▸▸ O percurso percorrido é memorizado.
5. Premir a tecla **M3** ▸/▸▸. O percurso memorizado é percorrido.

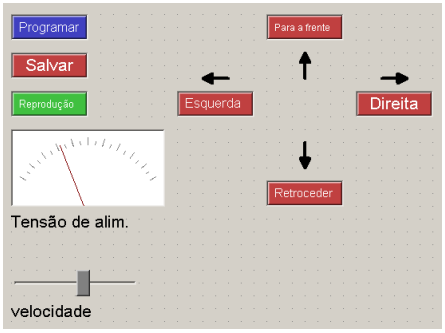
Com uma aplicação deste gênero, a programação de robôs se transforma numa brincadeira de crianças! Tem de ter em atenção, que o percurso memorizado é eliminado, assim que pára o programa através do botão Prog. na interface.

## ROBO RF Data Link

■ A interface aérea ROBO RF Data Link, n.º de art. 93295 substitui o cabo da interface entre PC e interface através de uma transmissão de dados sem fios. Isso é ótimo. Primeiro não é necessário conectar e voltar a desconectar o cabo toda vez que carrega um programa na interface. E em segundo lugar, pode operar programas sem cabos, no modo online, e, deste modo, detectar erros muito mais facilmente do que no modo download. E, finalmente, é possível comandar os robôs móveis no modo online, através de um painel de comando no ROBO Pro, de modo semelhante ao controle remoto IR online, através da tela. Ao contrário do controle remoto, é possível ainda visualizar adicionalmente na tela dados, que a interface fornece, p. ex. valores de variáveis ou entradas analógicas, tensão de alimentação, que a bateria fornece, velocidade dos motores.

Como exemplo, reconstruímos o programa Teach-In e comandamos o robô rolante através de um painel de comando. O programa se chama Mobile-Teach-RF.rpp. Também o pode experimentar através do cabo de interface. Isso é, no entanto, extremamente desconfortável. O raio de ação do modelo fica limitado, o cabo se enrola e o robô deixa de conseguir virar corretamente. Com certeza que vai querer adquirir imediatamente o RF Data Link.

Carregue o programa Mobile-Teach-RF.rpp.
Mude na barra de funções do programa principal para painel de comando. A seguir inicie o programa no modo online. Agora pode comandar e programar o modelo através dos botões no painel de comando.




1. Premir a tecla "Programar" O "processo de programação" é iniciado.
2. Comandar o modelo na direção pretendida, através das teclas de setas.
3. Premir a tecla "Salvar" O percurso percorrido é memorizado.
4. Premir a tecla "Reprodução" O percurso memorizado é percorrido.

O percurso memorizado também se perde aqui, quando o programa termina.

Obtém mais informações sobre a criação de painéis de comando no manual de instruções do ROBO Pro.

## ROBO I/O-Extension

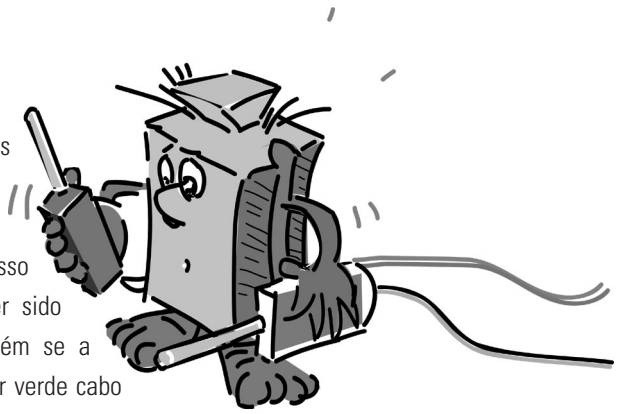
■ Caso construa um modelo com tantos sensores e motores, que as entradas e saídas da interface ROBO não sejam suficientes, pode conectar uma ROBO I/O-Extension, n.º de art. 93294 à interface. Desse modo, ficam disponíveis mais 8 entradas digitais, 4 saídas de motor e uma entrada de resistência analógica. Nesta extensão I/O pode conectar um segundo módulo, e conectar no segundo, um terceiro módulo, que serão depois todos comandados por uma ROBO Interface. Ficam então disponíveis, na totalidade, 16 saídas de motor, 32 entradas digitais, 5 entradas de resistência analógicas, 2 entradas de tensão analógicas, bem como 2 entradas para sensores de distância.

Se ainda não for suficiente, pode ainda comandar várias interfaces no PC, no Modo online, p. ex. uma numa interface serial COM, uma na porta USB ou 2 interfaces na porta USB, cada uma com até 3 ROBO I/O-Extensions! Isso pode se tornar muito confuso. O funcionamento de tudo isto se encontra descrito no capítulo 6 do manual de instruções ROBO Pro.

# Busca de falhas

■ Experimentar é divertido. Mas somente desde que tudo funcione. Na maioria das vezes isso acontece. Mas, infelizmente, nem sempre.
Quando um modelo não funciona, se questiona em primeiro lugar, se o mecanismo foi entendido correta e completamente e se a falha é imediatamente encontrada.
Em caso de falhas mecânicas, é possível ainda ver (montagem incorreta) ou sentir (falta de mobilidade) mais alguma coisa. Se surgirem ainda problemas elétricos, é mais complicado.
Os profissionais utilizam uma série de diferentes instrumentos de medição para a busca de falhas, como p. ex. medidor de tensão ou oscilógrafo. Nem todos têm acesso a este tipo de aparelhos. Pretendemos por isso tentar isolar e eliminar uma falha com meios mais simples.

**Montagem dos cabos**



Antes de começarmos com as nossas experiências, temos de terminar ainda alguns componentes dos módulos da Fischertechnik. P. ex. os conectores fornecidos devem ser conectados ás respectivas seções de cabos.
Primeiro o cabo é cortado. Medimos para isso os comprimentos indicados e cortamos as seções. Cada cabo é medido após estar completo. Para isso necessitamos da bateria e da lâmpada. Se a lâmpada acender, após ter sido conectada com a bateria, o cabo está em condições. Verificamos também se a atribuição de cores está correta, conector vermelho cabo vermelho, conector verde cabo verde.

**Teste de interface**

Se o programa (também o fornecido) não trabalhar juntamente com o nosso modelo, iniciamos o teste de interface. Este programa de ajuda permite-nos testar as entradas e saídas separadamente. Os sensores funcionam? Os motores rodam na direção correta? Os motores estão, em todos os nossos robôs móveis, ligados de tal forma que, em caso de sentido de rotação=esquerda, a roda ou a perna se movimenta para a frente. Se aqui se encontrar tudo em condições, procuramos a causa mecânica.

**Maus contatos**

Maus contatos são erros freqüentes. Por um lado, os conectores adaptadores podem estar soltos no plugue. Se for esse o caso, as molas do contato do conector são alargadas com uma pequena chave de parafusos. Cuidado, alargar demasiado provoca a quebra dos contatos ou conduz a uma falta de mobilidade ao encaixar.
Outra causa para os maus contatos são os acoplamentos de aperto soltos, nos locais de aparafusamento do conector. Por favor aparafusar cuidadosamente! Na ocasião é igualmente verificado, se nenhum dos fiozinhos de cobre finos estão partidos.

**Curto-circuitos**

Também poderia acontecer que seja provocado um curto-circuito devido à colocação errada dos cabos. Nesse caso, tudo deixa de funcionar como devia. Na bateria está integrado um fusível, que, em caso de corrente ou temperatura demasiado elevada, desliga a corrente. As saídas da interface são também desligadas, em caso de sobreaquecimento.
Pode também ser provocado curto-circuito, se não se apertar corretamente os pequenos parafusos nas fichas elétricas, com os quais o cabo é apertado. Estes ficam eventualmente a sobressair do bordo da

ficha. Se forem introduzidas duas fichas em dois plugues que se encontram diretamente ao lado um do outro na interface, de tal modo que dois dos pequenos parafusos se toquem, acontece um curto-circuito. Por isso, os pequenos parafusos devem ser sempre bem apertados e a ficha deve ser inserida de tal forma, que os parafusos não possam entrar em contato uns com os outros.

**Alimentação de corrente**

Se acontecerem falhas inexplicáveis durante o funcionamento, a causa pode ser a bateria vazia. A tensão desce brevemente ao conectar uma carga (motor ligado) e é ativado um reinicializar para o processador na interface. Ao acender o LED vermelho, a interface ROBO avisa quando a tensão da alimentação de corrente é demasiado reduzida. A bateria necessita então de ser carregada.

**Erro de programação**

Se, nos programas por si elaborados, surgirem erros que não são possíveis de explicar, deve, por razões de segurança, ser executado um programa o mais parecido possível com os fornecidos, de modo a que seja possível excluir avarias elétricas ou mecânicas. No Modo online é possível seguir o fluxo do programa na tela. Se o programa chegar a um determinado ponto e não continuar, é possível procurar aqui a causa, p. ex. entrada ou motor incorretos selecionados, valor incorreto consultado numa ramificação ou ligações J/N trocadas.

Se nada disto tiver sucesso, resta contatar o serviço de assistência fischertechnik (e-mail: info@fischertechnik.de).

Ou então visite o nosso site na Internet, em www.fischertechnik.de. Lá tem acesso a um fórum, chat, mercado, galeria e pode se tornar membro do Fischertechnik Fanclub gratuitamente.

Desejamos a você ainda muitas horas de divertimento com o ROBO-Mobile-Set com vários efeitos Aha.